

Remarks

Following the above amendments, claims 1-20 are pending in this application. The examiner has rejected claims 1-11 and 13-20 as being obvious under 35 U.S.C. 103(a) over U.S. Patent No. 6,289,396 to Keller in view of alleged applicant admitted prior art and U.S. Patent No. 6,425,038 to Sprecher. The examiner has additionally rejected claim 12 under 35 U.S.C. 103(a) over U.S. Patent No. 6,289,396 to Keller in view of alleged applicant admitted prior art; Sprecher; and U.S. Patent No. 5,754,858 to Broman.

A. Sprecher Does Not Establish a Prima Facie Case of Obviousness

Applicants respectfully submit that a prima facie case of obviousness has not been established and that a rejection of the pending claims on obviousness grounds is improper. In particular, Sprecher does not disclose those elements of the claims that are not present in the claims of the pending patent application. In particular, Sprecher does not disclose a “service layer” that is compiled against the kernel of the operating system, and Sprecher does not disclose a service layer that is compiled against the kernel of the operating system following a modification to the kernel of the operating system.

A prima facie case of obviousness requires a showing that all of the claim limitations of the rejected claims are taught or suggested by the prior art. Manual of Patent Examining Procedure 2143 and 2143.03. The establishment of a prima facie case of obviousness requires that *all* the claim limitations be taught or suggested by the prior art. MPEP 2143.01 (emphasis added). “All words of a claim must be considered in judging the patentability of that claim against the prior art.” *In re Wilson*, 424 F.2d 1382, 1385, 165 U.S.P.Q. 494, 496 (CCPA 1970).

B. Sprecher Does Not Disclose a Service Layer That is Compiled Against the Kernel of the Operating System Following a Modification to the Kernel

According to the examiner, Sprecher teaches the compiling of the service layer against the kernel of the operating system in column 5, lines 17-24, which are set out in their entirety below:

For this reason, the operating system interrupt router 38 may then call a compiled interrupt service routine 40 (per arrow 37) which polls the hardware cards 14 indicated by arrows 42 to identify the particular hardware cards 14 requiring service. The compiled interrupt service routine 40 must be fixed at the time of generation to the operating system kernel, and cannot be easily replaced after the kernel is prepared and embedded in the industrial controller 10.

(Sprecher, col. 5, l. 17-24). As part of the rejection, the examiner equates the interrupt service routine 40 of Sprecher to the service layer of the claims. Each independent claim of the present application has been amended to specify that the service layer is compiled against the kernel of the operating system following each modification to the kernel of the operating system. There is no suggestion in Sprecher that the interrupt service routine of Sprecher can be compiled against the kernel of the operating system following each modification to the kernel of the operating system.

Instead, Sprecher confirms that the interrupt service routine is fixed. “The compiled interrupt service routine 40 *must be fixed* at time of generation to the operating system kernel, and *cannot be easily replaced*” This language plainly contemplates that the interrupt service routine cannot be modified. In contrast, the service layer of the amended claims is recompiled following each modification to the open source kernel of the operating system. Because the open source kernel of the operating system is easily modifiable, the service layer is recompiled following a modification to the kernel to match the compiled service layer to the

modified kernel. The ability to easily compile the service layer against a modified kernel is discussed in the Summary of the Invention in the present application:

Another technical advantage of the present invention is a method for distributing device driver software to users of a computer system having an open source operating system. A device driver having an open source service layer is provided. The provision of the open source service layer permits a user to modify or replace the kernel of the operating system. Once the user modifies the kernel of the operating system, the user can rebuild the device driver of the computer system by compiling the service layer against the modified operating system kernel. In this manner, the user can continue to modify the open source operating system, while proprietary information of the computer system manufacturer is protected against disclosure and resides in the precompiled driver modules of the device driver.

(Application, page 5, l. 3-11). Thus, the advantages of recompiling the service layer following each modification to the kernel of the operating system are plainly stated in the application. Sprecher does not disclose this ability. Rather, Sprecher discloses an interrupt service routine that content “must be fixed” and “cannot easily be replaced.”

C. Sprecher Teaches Away From the Invention

Because Sprecher plainly discloses that the interrupt service routine of Sprecher “must be fixed” and “cannot easily be replaced,” it is plain that Sprecher teaches away from the invention of the amended claims, which require the compilation of the service layer following each modification of the kernel. A reference teaches away from the invention when a person of ordinary skill in the art, upon reading the reference, would be led down a path that is divergent from the path of the patent applicant. *See Tec Air, Inc. v. Denso Mfg. Mich. Inc.*, 192 F.3d 1353 (Fed. Cir. 1999); *In re Gurley*, 27 F.3d 551, 553 (Fed. Cir. 1994) (explaining that a reference teaches away if it suggests a line of development that is unlikely to produce the result sought by

the applicant). It is improper to combine references that teach away from their combination. *In re Grasselli*, 713 F.2d 731 (Fed. Cir. 1983); MPEP 2145.

Here, Sprecher is an improper candidate for combination in an obviousness rejection because Sprecher teaches away from the claimed invention. Sprecher notes that the interrupt service routine “must be fixed.” In contrast, the claims of the present application clarify that the service layer is compiled following each modification to the kernel of the operating system. Sprecher actually teaches away from this disclosure.

Applicants respectfully submit that Sprecher does not disclose the teaching of compiling the service layer following each modification to the kernel of the operating system. In addition, Sprecher actually teaches away from the claimed invention by specifying that the interrupt service routine of Sprecher “must be fixed” and “cannot be easily replaced.” Applicants respectfully submit that a prima facie case of obviousness has not been established with respect to independent claims 1, 8, and 13.

D. Dependent Claims 2-7, 9-12, and 14-20

Dependent claims 2-7, 9-12, and 14-20 will not be discussed individually herein, as each of these dependent claims depends, either directly or indirectly, from an otherwise allowable base claim.

Conclusion

The applicants respectfully submit that pending claims 1-20 of the present invention are allowable. The applicants respectfully request that these claims be passed to issuance.

Respectfully submitted,



Roger Fulghum

Registration No. 39,678

Baker Botts L.L.P.
910 Louisiana
One Shell Plaza
Houston, Texas 77002-4995
(713) 229-1707

Baker Botts Docket Number: 016295.0732

Date: October 3, 2005